

CLAIMS

What we claim is:

1 1. An apparatus comprising a code block alignment system that reduces a number of
2 byte-level file differences between an original file and a new file by sorting and moving
3 code blocks of the original file in order to align the code blocks of the original file in the
4 same order as code blocks of the new file, and dynamically recording information of the
5 block movements in a list and encoding the list.

1 2. An apparatus comprising:
2 means for receiving an original file and a new file, wherein the new file includes
3 an updated version of the original file;
4 means for determining an order of code blocks of the new file using index values;
5 means for sorting code blocks of the original file and generating a largest
6 increasing subsequence (LIS) of code blocks according to the index values;
7 means for generating lists of original order numbers of the code blocks of the
8 original file affected by code block movements; and
9 means for moving the code blocks of the original file to locations in the original
10 file according to the largest increasing subsequence of code blocks, wherein the code
11 blocks of the original file are aligned in the same order as code blocks of the new file.

1 3. A system for updating electronic files of remote devices, comprising:
2 a first device including a file differencing engine that generates differences
3 between an original version and a new version of an electronic file by:
4 determining an order of code blocks of the new version using index
5 values;
6 sorting code blocks of the original version and generating a largest
7 increasing subsequence (LIS) of code blocks according to the index values;
8 generating lists of original order numbers of the code blocks of the
9 original version affected by code block movements;

10 moving the code blocks of the original version to locations in the original
11 version according to the largest increasing subsequence of code blocks, wherein
12 the code blocks of the original version are aligned in the same order as code
13 blocks of the new version;
14 generating an encoded list including information of the code block moves;
15 transmitting the encoded list to a second device; and
16 a file updating engine hosted on the second device, the file updating engine
17 generating a copy of the new version using a difference file and information of the code
18 block moves.

1 4. The system of claim 3, wherein the second device includes at least one of cellular
2 telephones, portable communication devices, personal digital assistants, personal
3 computers, and portable processor-based devices.

1 5. A method for generating difference files, comprising:
2 receiving an original file and a new file, wherein the new file includes an updated
3 version of the original file;
4 determining an order of code blocks of the new file using index values;
5 sorting code blocks of the original file and generating a largest increasing
6 subsequence (LIS) of code blocks according to the index values;
7 generating lists of original order numbers of the code blocks of the original file
8 affected by code block movements;
9 moving the code blocks of the original file to locations in the original file
10 according to the largest increasing subsequence of code blocks, wherein the code blocks
11 of the original file are aligned in the same order as code blocks of the new file; and
12 generating an encoded list including information of the code block moves.

1 6. The method of claim 5, further comprising:
2 generating mappings between code blocks of the original file and corresponding
3 code blocks of the new file; and
4 generating a mapping array using information of the mappings.

1 7. The method of claim 6, further comprising:

2 generating a longest increasing subsequence (LIS) of the mapping array;
3 identify membership of the LIS for each code block of the original file; and
4 generating a membership array using information of the LIS membership.

1 8. The method of claim 5, further comprising receiving at least one of a mapping
2 array, a membership array, a start address array, a size array, and an original code block
3 order number array, wherein the mapping array includes information of mappings
4 between code blocks of the original file and corresponding code blocks of the new file,
5 wherein the membership array includes information of a membership of a longest
6 increasing subsequence (LIS) for each code block of the original file, wherein the start
7 address array includes information of code block start addresses, wherein the size array
8 includes information of code block sizes, and wherein the original code block order
9 number array includes information of the original order of code blocks in the original file
10 prior to the code block movements.

1 9. The method of claim 5, further comprising:

2 transferring the encoded list to at least one remote system via at least one
3 coupling, wherein the remote system includes at least one processor-based system,
4 wherein the coupling includes at least one of a wireless coupling, a wired coupling, and a
5 hybrid wireless/wired coupling; and
6 updating a hosted copy of the original file in the remote processing systems using
7 a difference file and the encoded list.

1 10. A method for reducing a number of changes between an original file and a new
2 file, comprising:

3 determining an order of code blocks of the new file using index values;
4 sorting code blocks of the original file and generating a largest increasing
5 subsequence (LIS) of code blocks according to the index values;

6 generating at least one list of original order numbers of the code blocks of the
7 original file affected by code block movement; and
8 moving the code blocks of the original file to locations in the original file
9 according to the largest increasing subsequence of code blocks, wherein the code blocks
10 of the original file are aligned in the same order as code blocks of the new file.

1 11. A computer readable medium including executable instructions which, when
2 executed in a processing system, reduce a number of changes between an original file and
3 a new file by:
4 determining an order of code blocks of the new file using index values;
5 sorting code blocks of the original file and generating a largest increasing
6 subsequence (LIS) of code blocks according to the index values;
7 generating lists of original order numbers of the code blocks of the original file
8 affected by code block movements; and
9 moving the code blocks of the original file to locations in the original file
10 according to the largest increasing subsequence of code blocks, wherein the code blocks
11 of the original file are aligned in the same order as code blocks of the new file.